# DHS Bulletin: Securing Control Systems

Cyber Security Research Department
Idaho National Laboratory
11 February 2005

## *Information:*

Control Systems (CS) manage the nation's Critical Infrastructure; therefore, it is paramount that secure systems be established. However, integrating security into control system environments is a much more inflexible process than in general IT networks. In lieu of this and the incredibly varied architecture of CS network architecture, control systems administrators and operators must carefully review the recommendations for securing control system networks before applying the changes. Testing and deployment of security configurations or updates should be performed on development, test, or backup systems and monitored carefully for impact before being put into practice on a production control system.

## *Security Principles:*

The first key principle of security is establishing the importance of each CS computer on the network. The more critical the system is to the network, the higher the level of security required.

The second key principle of security is to limit the services and resources on the system to those processes that are absolutely necessary. Implement the required services and periodically verify if those services are still required.

The third key principle of security is to limit user access to the services and resources on the network and its individual components. If a user, application, or service no longer requires access, remove them from the access lists. Evaluate user lists and application permissions regularly to ensure they have appropriate levels of access.

## *Security Methodology:*

There are three parts of a control system that must be secured:

1. Its network communications,
2. The base operating system of each host on the network, and
3. The CS applications themselves.

Each of these components of the system must be individually evaluated and locked down in order to achieve the desired level of security.

Once a vulnerability has been identified, CS administrators have five methods of closing the hole. The first four are generally technical or technological solutions while the fifth is policy-based.

1. Blocking access to resources and services—this technique is generally employed on the network through the use of perimeter routers with ACL lists, firewalls, or proxy servers. It can be enabled on the host via host-based firewalls and anti-virus software.
2. Detecting malicious activity—detection can be network or host based and requires regular monitoring of logs by experienced administrators. Intrusion detection systems (IDS) are the common means of identifying problems on a network, but they can be deployed on individual hosts as well. Auditing and event logs should be enabled on individual hosts when possible.
3. Mitigating possible attacks—Mitigation allows administrators to control access to a vulnerability in such a fashion that the vulnerability cannot be exploited. This can be done by enabling technical workarounds, setting up filters, or running services and applications with specific configurations (change default setting).
4. Fixing core problems—Resolution of core security problems almost always requires updating, upgrading, or patching the software vulnerability or removing the vulnerable application. The software hole can be resident in any of the three layers (networking, operating system, or application) and the medication should be provided by the vendor or developer for administrators to apply.
5. Security policies—Security policies should be developed for the control system network and its individual components, but they should be reviewed periodically to reflect the current threat environment, system functionality, and required level of security.

## *Specific Recommendations (SCADA Focus):*

- Restrict ICCP links so they are allowed to carry only ICCP data. CS-to-CS communication links are not as tightly secured as links between the business and CS networks. Breaking into a CS network using these links is often the easiest way to get onto a control system network, and transmitting exploit information via ICCP traffic is an easy way to take advantage of the communications. (Network, blocking)
- Mirror router and firewall ACL rules to reduce the chance of misconfiguration and help control versions of rules deployed. (Network, blocking)
- Restrict outbound traffic from the control system network. Modern attackers utilize client-side attacks by piggybacking on existing communications. They then make the trusted device on the network call back out through the firewall, effectively bypassing security controls. (Network, blocking)
- Encrypt data being passed from the CS to the business network to prevent attackers from accessing and manipulating traffic between the two networks. (Network, blocking)

- Hard code ARP tables to prevent ARP table poisoning, the most popular way to manipulate insecure protocols. While this technique is not feasible on a business network, the limited number of hosts on a control system network can be effectively protected this way.
- Use host tables rather than DNS so DNS cannot be manipulated to gain control of hosts and network communications. (Network, blocking)
- Use personal firewalls on individual hosts and configure the rule sets to limit access to and from the host. (Network, blocking)
- Filter or disable support for RIPv1 packet redirects because they are rarely used on a production network. Validate due to specific sector use. (Network, blocking)
- Filter or disable ICMP packet redirects because they are not widely used on CS networks (validate). (Network, blocking)
- Disable Windows domains or NIS on UNIX and Linux environments when possible because they are popular targets for attack and inherently insecure. (Network, blocking)
- Remove any unnecessary, default routes that lead back to the firewall and then to other networks. If hosts do not need to communicate with other networks, removing the routing information will not disrupt functionality and prevents the machine from calling home if it is successfully compromised. (Network, blocking)
- Disable proxy ARP features on routers so they can't be used by internal machines to discover routes off the CS network without a routing table. (Network, blocking)
- Periodically checksum and account for files to identify potential problems with missing or altered files. (Policy, detection)
- Use separate hosts to log data and control processes to add another layer of defense on the network. Restricting communications between the logging host and the control host makes it more difficult for an attacker to control processes. (Policy, blocking)

## Stuff you should already know:

- Session tracking firewalls - Router ACLs are usually not enough to keep out a skilled attacker. Installing a session tracking firewall is a must.
- Don't share DNS, Wins, Domain, and NIS environments - If any of these protocols traverse your SCADA LAN firewall, they can be manipulated to give the attacker control. Pay special attention to how patches are downloaded and installed.
- Use SSH instead of RSH and Rlogin - RSH and RLogin are easily used to access even by unskilled attackers. SSH can be used as a replacement for the r-services in almost all instances with a little configuration.
- Close unused ports - If you don't need it, shut it off. It gives the attacker less to talk to.

- Restrict .rshosts file to only those that need it - If r-services need to be used, general entries such as "+ root" should be removed.
- Disable NFS unless absolutely necessary - Run NFS over SSH if needed.
- NFS has long been the target of attackers. The most modern versions have less vulnerability, but as still architecturally bad. Tunneling NFS over SSH removes most of the vulnerabilities.
- Use shadowed passwords on UNIX machines - All modern UNIX flavors support shadowed passwords. Without this feature any user has access to the password hashes.
- Don't allow port 6000 from the network - If an attacker can access X11 sessions, he can manipulate the display, record keystrokes and a host of other fun things. If port 6000 is utilized for specific processes, utilized IDS for monitoring traffic for potential attacks.
- Aggregate logs to a central location - An attacker can usually clean up the logs on a host he's compromised. Logging to a central server can often preserve evidence of tampering attempts.
- Monitor and not abnormal log sizes - Attacker often have to try many combinations to control a machine. Those attempts often increase the size of logs. Larger than normal log files should be investigated.